AD-A042 138 SYSTEMS CONSULTANTS INC WASHINGTON D C F/G 17/4
ANALYSIS OF SRBOC WORLDWIDE CHAFF CARTRIDGE PROVISIONING REQUIR--ETC(U)
JUN 77 S C SCHIFF N60921-77-C-0089
UNCLASSIFIED SCI-6165-77-006-VOL-1 NL

| OF |

ADA042138

END
DATE
FILMED
8-77

SCI-6165-77-006

(1)

B.S.

ANALYSIS OF SRBOC WORLDWIDE CHAFF CARTRIDGE PROVISIONING REQUIREMENTS

VOLUME I: RF DECOY PLACEMENT SIMULATION (RFDPS) DESCRIPTION AND USE

8 June 1977

Prepared for

COMMANDER, NAVAL SURFACE WEAPONS CENTER,
WHITE OAK LABORATORY
WE-202

Under Contract Number
N60921-77-C-0089

Prepared by

Systems Consultants, Inc.
1054 31st Street, N.W.
Washington, D.C. 20007

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Chaff Cartridge	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) Analysis of SRBOC Worldwide Provisioning Requirements, Volume I. RF Decoy Placement Simulation (RFDPS) Description and Use.		5. TYPE OF REPORT & PERIOD COVERED (9) Technical Report.
		6. PERFORMING ORG. REPORT NUMBER (14) SCI-6165-77-006-Vol-1
7. AUTHOR(s) (10) S.C. Schiff		8. CONTRACT OR GRANT NUMBER(s) (15) N60921-77-C-0089 New
9. PERFORMING ORGANIZATION NAME AND ADDRESS Systems Consultants, Inc. 1054 31st Street, N.W. Washington, D.C. 20007		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Surface Weapons Center White Oak Laboratory Silver Spring, MD 20910 Attn: WE-05		12. REPORT DATE (11) 8 June 1977
		13. NUMBER OF PAGES 43
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) (12) 45p.		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release: Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Chaff Countermeasures
Antiship Missile Defense
Decoys
Digital Simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Chaff decoy placement and related antiship missile defense problems have been analysed by means of digital simulation. This document describes the simulations in terms of mathematical models, evaluation criteria and software.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

388 950 LB

I. INTRODUCTION

This report describes a digital simulation for the evaluation of chaff decoy placement and related antiship missile defense problems. The simulation is the latest in a series of programs developed by Systems Consultants, Inc. (SCI) for the U.S. Navy; this one was developed for the Naval Surface Weapons Center White Oak Laboratories under contract N60921-77-C-0089. The purpose of that contract was to investigate logistic requirements for the MK 36 and MK 182 Super Rapid Bloom Offboard Chaff (SRBOC) decoy system.

This document represents the opinion of Systems Consultants, Inc. The views and conclusions set forth herein are those of the author and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Naval Surface Weapons Center or the U.S. Government.

II. OVERVIEW OF THE SIMULATION

A. BACKGROUND

A decoy is a device which is designed to present an appearance which is indistinguishable from that of a true target when viewed by a specified sensor, so that deception of the sensor or its operator can be accomplished. The degree to which the decoy must resemble the true target, in terms of all of its measurable characteristics, depends upon the attributes of the sensor. For example, a sensor which measures both the signature and motion characteristics of its target cannot be reliably deceived by a decoy which differs significantly from a true target in either respect.

Decoys have been in use since the earliest days of radar. The ways in which decoys can be used to defeat weapons correspond to the sequence of operations of weapon targeting. During World War II, for example, chaff decoys were used by the allies against German air defenses in two ways: corridor chaff was employed to mask entire flights of bombers to deny information on the number of aircraft involved; chaff was also used to degrade the tracking capabilities of the German gun-laying radars. Currently, decoys are being developed to counter antiship missiles (ASMs). Against these, there can be as many as three modes of decoy operation:

(1) Target Selection Confusion of the Operator (TSCO), wherein the purpose of the decoy is to cause the weapon to be launched against a decoy rather than a true target or to deny targeting data.

(2) Initial Acquisition Decoying. Many guided weapons, particularly long range ones, do not track their targets continuously from launch, but fly under the control of an autopilot for some specified period, after which the

target must be acquired. The purpose of initial acquisition (IA) decoying is to cause the missile to acquire a decoy rather than a true target.

(3) Break Lock Decoying. Once a missile has acquired a true target, break lock (BL) decoys can be used to degrade the tracking accuracy of the missile.

The foregoing example thus represents instances of TSCO and BL decoying.

The efficacy of a decoy depends upon satisfaction of two criteria: indistinguishability from a true target and placement relative to the true target and the sensor. The decoy must be indistinguishable from a true target in the sense that the victim sensor cannot reject the decoy on the basis of received data. The decoy's position is important because sensors accept only those signals which emanate from the volume of space where true targets are expected to be. Sensor characteristics may differ according to their mode of operation (pre-launch targeting, acquisition of the target by the weapon, and terminal homing) and so the requirements for decoy characteristics and placement may also vary according to decoy mode (TSCO, IA or BL). Since both the indistinguishability and placement criteria must be satisfied for the decoy to be effective, they can be evaluated independently in most instances. This principle is central to the utility of the digital simulation described in this report, which is used to evaluate decoy placement in the BL mode against radar guided ASM.

The decoy placement criteria can always be satisfied if there is sufficient selectability in decoy placement relative to the defended ship (true target) and the victim, and if the timing of the decoy launch is appropriate. However, practical limitations to decoy placement will

always exist. Some of these limitations arise as a result of uncertainity as to the location (range, altitude and direction), speed, direction of motion and detailed characteristics of the ASM. Others result from the design limitations of the decoy system: in order to minimize equipment cost and size, shipboard decoy launchers employ fixed train and elevation angles; the decoys themselves are ballistic and use a single propulsion system. Consequently, the design of decoy systems to achieve maximum placement effectiveness and the evaluation of the placement effectiveness of a decoy system are relatively complicated problems. The simulation described subsequently in this report was developed specifically to address these problems.

B. SCOPE

Decoy effectiveness against a particular ASM depends upon the decoy placement geometry and also upon the performance characteristics of the decoy (e.g., its radar cross section). Consequently, it is possible to determine decoy effectiveness only through consideration of both performance and placement. Since these are independent of each other for most applications (and in particular for the threat model used herein) it is possible to write the following expression for decoy effectiveness:

$$\eta = \eta_{\text{placement}} \quad \eta_{\text{performance}}$$

where $\eta_{\text{placement}}$ is the placement effectiveness and $\eta_{\text{performance}}$ is the performance effectiveness, i.e. the probability that the missile will home on the decoy given that the decoy is placed within its field of view. The decoy placement simulations have been designed to calculate $\eta_{\text{placement}}$, and therefore yield η, the decoy effectiveness, only under the condition that $\eta_{\text{performance}}$ is equal to unity. That is, the simulation gives the upper bound on decoy effectiveness.

The results from the simulation are further limited because of designed-in assumptions about the nature of the engagement. Placement effectiveness depends, inter-alia, upon the range and direction of the threat at the time of decoy launch, the speed of the ship, and the speed and direction of the true wind. The simulation yields average placement effectiveness values for a given ship speed, decoy launcher, and decoy against a specified threat by sampling from a three dimensional space which is bounded by minimum and maximum threat directions, wind directions, and wind speeds. The first of these is uniformly distributed, i.e. any value between the bounds is equally likely. The wind conditions (true speed and true direction) can be chosen in either of two ways. The preferred method is to choose wind direction from a uniform population, and to select the speed from a parent population selected to closely match empirical data (Long et. al., 1965). The second method involves successively incrementing true wind speed or true wind direction on a case-by-case basis, so as to arrive at a sample which includes a predetermined matrix of true wind speed and true wind direction values. For obvious reasons, this option is referred to as determistic. As a result of the sampling method, the average results of the simulation cannot be applied to any one scenario, but represent the placement effectiveness achieved over a large number of scenarios which occur under varying conditions. Since the simulation has provisions for detailed reports giving the placement results on a sample by sample basis, it can also be used in evaluating decoy placement performance in any particular scenario.

C. STRUCTURE

The simulation represents the embodiment of mathematical models and evaluation criteria in software. There are three elements to the mathematical model: the decoy, ship and threat. The mathematical model describes the salient features of each element and their motion relative

to each other. The evaluation criteria consist of sets of precise conditional statements, the outcomes of which are classified as pass or fail with respect to various success determinants. Implemented in software, the mathematical model and evaluation criteria are used with statistically independent samples of the appropriate variables to perform a time line analysis of an engagement. By manifold replication of this process, average placement effectiveness values are obtained (Monte Carlo method). The software also includes report generators of two types: summary and detailed. The summary reports yield aggregated results (e.g. average placement effectiveness) while the optional detailed reports yield results (values of the random variables and pass/fail indication) on a case by case basis.

The mathematical model upon which the simulation is based is discussed in Section III of this report. Section IV describes the evaluation criteria, and Section V includes FORTRAN IV software and interactive user's guide for the simulation program.

III. THE MODEL

The simulation includes sets of equations for the coordinates of the ship, decoy and threat in two reference frames: the earth frame (so called because it is at rest on the surface of the earth) and the missile frame (which is at rest with respect to the missile). These reference frames are connected by means of a time dependent coordinate transformation: at any instant the transformation consists of a translation plus a rotation of coordinates. Once the transformation is effected, the coordinates of any point are given in both frames. By referencing the missile frame to its line of flight direction, the angles from boresight to any point in space can be determined, that is, it is deteminable whether a point lies within the missile seeker antenna beamwidth. Naturally, it is also easy to determine the angle subtended by an object, and also distances. In the following subsections the equations of motion for the ship, decoy and missile are developed. Field of view and range considerations are discussed in section IV, as they are naturally related to the evaluation criteria.

A. SHIP

For the purposes of these simulations the ship is represented as a line of length 2BSHD (Bow-Stern Half Distance). Its motion is uniform in the earth frame, that is, it travels at a constant speed in a constant direction. The reference direction is chosen to be along the x-axis of the earth frame; at time T=0 the midpoint of the ship is located at the origin of the coordinate system, as shown in Figure (1). (In the figure, the vertical

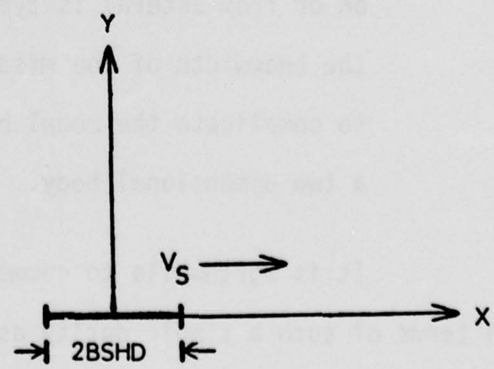

FIGURE 1

direction is normal to the page). The equations of motion for the ship's midpoint are thus:

$$X_S = V_S T$$
$$Y_S = Z_S = 0 \quad (1)$$

while the coordinates of the bow and stern are shifted along the x-axis by +BSHD and -BSHD, respectively;

$$X_{SB} = X_S + \text{BSHD} \qquad (Y_{SB} = Z_{SB} = 0)$$
$$X_{SS} = X_S - \text{BSHD} \qquad (Y_{SS} = Z_{SS} = 0) \quad (2)$$

It has been determined that the ship can be accuately modeled in terms of a line rather than a three dimensional body for the following reasons:

(1) The missile track point is generally within a few meters of the ocean surface, so that it is not necessary to represent the ship in terms of a vertical extent.

(2) The ship is very long in comparison to its width (typical destroyers have L/B ratios of about 9), so as a consequence the width of a ship contributes little to its angular extent when viewed from distances of practical interest. Further, at these distances the angle subtended by the ship's width (when viewed bow-on or from astern) is typically very much smaller than the beamwidth of the missile. Thus it is not necessary to complicate the model by treating the ship as even a two dimensional body.

It is worthwhile to comment that the modeling of the ship even in terms of such a simple entity as a line has very little effect upon the outcome of the simulation analysis except in the case where a very high

performance missile (narrow beamwidth and range gate) is assumed; this applies likewise to the decoy. The initial simulation which was used by SCI to evalaute decoy placement effectiveness represented both the ship and the decoy as points, yet exhibits the same sort of qualitative behavior as does this one with respect to the more fundamental of the success criteria.

B. DECOY

A decoy is treated within the model as an object which in effect comes into existence at some precalculated time and position relative to the ship. This reflects the need to model the times of flight and cross section development of the decoy, (These times are called T_f and T_b). Suppose that the decoy is launched at time T=0 from a launcher located a distance d_L forward of amidships and pointed at an angle ϕ off the bow, as shown in Figure 2. (Note that angles are measured according to the mathematical, not nautical, convention.) Let R_C be the horizontal distance of decoy deployment and h_o be the altitude of decoy deployment. Since the decoy initially has a velocity component V_S in the X direction, its position at time $T=T_f$ will be given by

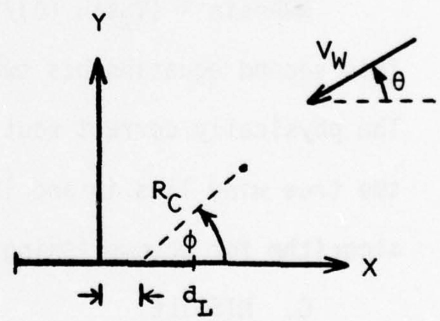

FIGURE 2

$$
\begin{aligned}
X_C &= d_L + R_C\cos\phi + V_S T_f \\
Y_C &= R_C \sin\phi \qquad (3) \\
Z_C &= h_o
\end{aligned}
$$

Since the decoy is airborne, it will be subject to vertical motion under the influence of gravity and horizontal motion under the influence of the true wind, as measured in the earth frame. Let θ be the angle of the true wind as measured off the x-axis in Figure 2 and let the true wind

speed by V_W. Suppose also that the decoy quickly reaches its terminal velocity of fall V_{CV}. Then the coordinates of the decoy at time $T>T_f$ are

$$X_C=d_L+R_C\cos\phi+V_ST_f-V_W T\cos\theta$$

$$Y_C=R_C\sin\phi-V_WT\sin\theta \qquad (4)$$

$$Z_C=h_o-V_{CV}(T-T_f)$$

The decoy is treated as a sphere of radius Dcc centered at the coordinates given by (4).

The horizontal motion of the decoy relative to the ship will be determined by the relative wind (wind speed and direction as measured by the ship - See Schiff, et. al., 1975). Since the relative wind conditions are of tactical importance, the calculation of relative wind speed and direction are included in the simulation. The relative wind speed is given by

$$V_{RW}=\sqrt{V^2{}_S+V^2{}_W+2V_SV_W\cos\theta}$$

while the direction is

$$RWA=\sin^{-1}(V_W\sin(\theta)/V_{RW})$$

This second equation has two admissible roots which differ by 180^o - 2RWA. The physically correct root is determined by considering which quadrant the true wind lies in and its speed relative to that of the ship. The algorithm for accomplishing this is discussed in Appendix A.

C. MISSILE

The missile model utilized in the simulation has been designed for maximum generality, to minimize reliance upon the need for detailed intelligence information. The general characteristics of the model are:

(1) Angle gating so as to process information emanating within a cone centered on the missile boresight, of central angle 2 BWHA (BeamWidth Half Angle).

(2) Range gating to further restrict the volume of space from which processed signals can emanate. The width of this

gate is 2RGHW (Range Gate Half Width). The range gate, like the angle gate, is assumed to be geometrically centered on the traget.

(3) Fixed ground speed V_T

(4) Fixed altitude Z_T

(5) Proportional (Pursuit) navigation in the azimuth plane (but see below!).

For reasons of economy of simulation design and execution, the flight of the missile is decomposed into two segments: prior to and subsequent to detection of the decoy. From its initial point the missile moves along a straight line toward the estimated point of impact with the ship midpoint until the decoy is detected. Upon satisfaction of the decoy detection criterion, the missile turns toward the decoy at a rate which is proportional to the azimuth angle off boresight to the decoy.

The two segments of the missile dynamic model represent different navigational modes: lead and proportional. This is in contrast to real missiles, which would employ a single navigational mode. However, the differences between the flight paths for the two modes depend in the main upon the relationship between the target and missile velocities. Since the missile speed is at least an order of magnitude greater than those of either the ship or the decoy, the difference between the flight profiles for the two modes is insignificant, and thus the simplification made in the model is entirely justified.

At time T=0 the missile is a distance R_0 from the ship at bearing γ. The missile is moving in the xy (horizontal) plane with speed

V_T, while the ship moves along the x-axis of the earth frame with speed V_S. Let T_e be the time of impact; the missile will move a distance $V_T T_e$ and the ship will move a distance $V_S T_e$ in the time interval (0,Te). Using the law of sines

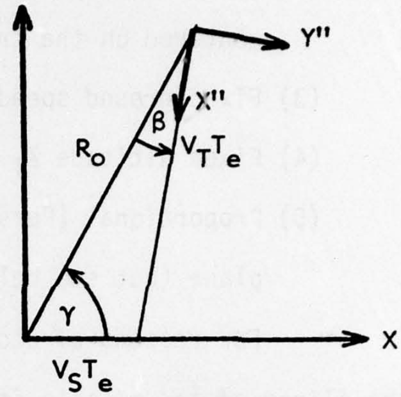

FIGURE 3

$$\beta = \sin^{-1}\left\{\frac{V_S \sin\gamma}{V_T}\right\} \qquad (5)$$

subject to $V_S \leq V_T$, and

$$T_e = \frac{R_O \sin\gamma}{V_T \sin(\beta+\gamma)} \qquad (6)$$

With these relationships the motion of the missile in the xy plane is given by

$$X_T = R_O \cos\gamma\,(1 - T/T_e) + V_S T$$

$$Y_T = R_O \sin\gamma\,(1 - T/T_e)$$

for times $T < T_e$.

The missile frame of reference is defined as that which is at rest with respect to the missile, with its origin corresponding to the missile location and its x-axis directed along the missile line of flight, as shown in Figure 3. The transformation between the earth frame and the missile frame is accomplished by first making a translation

$$X' = X - X_T$$

$$Y' = Y - Y_T$$

followed by a rotation in a counterclockwise direction

$$X'' = -X' \cos(\beta+\gamma) - Y' \sin(\beta+\gamma)$$

$$Y'' = X' \sin(\beta+\gamma) - Y' \cos(\beta+\gamma)$$

or

$$X'' = (X_T - X)\cos(\beta+\gamma) + (Y_T - Y)\sin(\beta+\gamma) \qquad (7)$$

$$Y'' = (X - X_T)\sin(\beta+\gamma) + (Y_T - Y)\cos(\beta+\gamma)$$

The coordinate transformation also includes a translation in the vertical direction:

$$Z''=Z-Z_T \tag{8}$$

During the decoy tracking segment of the missile flight, the missile trajectory in the horizontal plane is determined by the position of the target relative to the missile line of flight. In particular, the direction and rate of turn in the azimuth plane are related to the angle off boresight by

$$\frac{d\alpha}{dT}=\dot{\alpha}=-\kappa\alpha \tag{9}$$

where $\dot{\alpha}$ is the rate of turn, α is the azimuth angle off boresight, and κ is the navigational system constant. Equation (9) describes a simple proportional navigation system. The constant κ is related to the acceleration limit of the airframe, and in turn to the sensor beamwidth.

Suppose at some instant t_o the missile is flying at speed V_t in the positive x-direction in some reference frame; at that time an azimuth error angle α exists. According to the navigation system equation (9), the missile line of flight is rotated thru the angle $\kappa\alpha\delta t$ during the infinitesimal time interval δt, so that at time $t = t_o + \delta t$ the x and y velocity components of the missile are

$$V_x(t_o+\delta t) = V_t \cos \kappa\alpha\delta t$$

$$V_y(t_o+\delta t) = V_t \sin \kappa\alpha\delta t$$

as indicated in Figure 4. Thus,

$$A_x = \lim_{\delta t\to 0} \frac{V_x(t_o+\delta t) - V_x(t_o)}{\delta t}$$

$$= \lim_{\delta t\to 0} V_t \frac{(\cos \kappa\alpha\delta t -1)}{\delta t} \quad =0$$

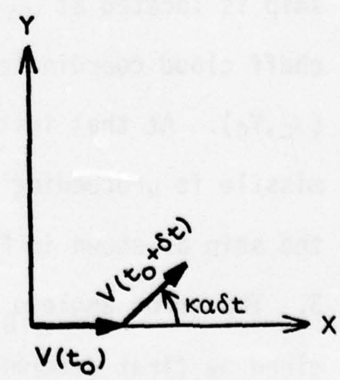

FIGURE 4

and $$Ay = \lim_{\delta t \to o} \frac{V_y(t_o+\delta t) - V_y(t_o)}{\delta t}$$

$$= \lim_{\delta t \to o} V_t \frac{\sin \kappa\alpha\delta t}{\delta t} = \kappa\alpha V_t$$

Consequently, if α_{max} is the maximum azimuth angle error which can be accepted and a_{max} is the airframe maneuver limit,

$$a_{max} = \kappa\, \alpha_{max}\, V_t$$

so that

$$\kappa = \frac{a_{max}}{\alpha_{max}\, V_t} \qquad (10)$$

In order to calculate κ, α_{max} is generally taken to be approximately equal to the scanned half beamwidth limits of the missile radar antenna. On the basis of some further simplifications, the required maneuver limit can be calculated based upon the missile field of view geometry so that κ can be uniquely determined on the basis of the missile speed, search beamwidth and minimum target acquisition range (Schiff, 1977).

The algorithm for the tracking of the decoy (subsequent to the break lock decision point) is as follows. At the moment of lock transfer, the missile coordinates in the earth frame are (X_t, Y_t); the ship is located at $(X_s, 0)$ and the chaff cloud coordinates are (X_c, Y_c). At that instant, the missile is proceeding toward the ship as shown in Figure 5. The error angle α_D is determined by first determining the orientation of the missile coordinate frame.

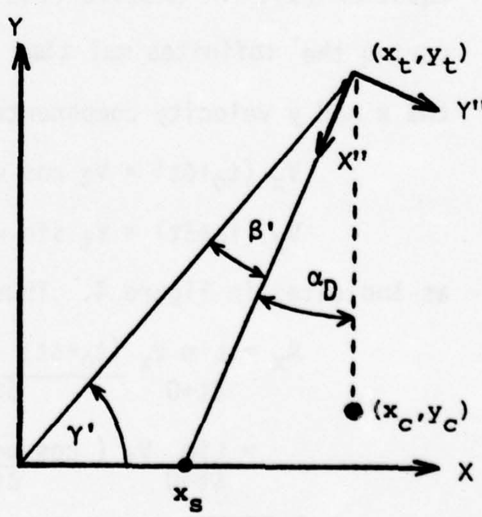

FIGURE 5

Now,

$$\gamma' = \tan^{-1} (Y_t/X_t) \quad (11)$$

and

$$\beta' = \sin^{-1} \left(X_s \cos\gamma' / \sqrt{(X_t - X_s)^2 + Y_t^2}\right)$$

The angle of rotation between the earth frame and the missile frame, for the missile in the first or second quadrants, is $180^o - (\beta' + \gamma')$. Accomplishing a translation of coordinates to (X_t, Y_t) followed by this rotation, the coordinates of the decoy are calculated. (Cf equation (7).) The error angle α_D is then given by: $\alpha_D = \tan^{-1} (Y_c''/X_c'')$ In the example of Figure 5, this angle is positive, and hence the first correction applied corresponds to a proper (counterclockwise) rotation in the missile frame. Were Y_c'' negative, the value of α_D would be negative and a clockwise rotation in the missile frame would be indicated to turn the missile line of flight toward the decoy.

Having determined α_D, the angle β' is changed according to

$$\beta'_{new} = \beta' + \kappa\alpha_D\delta t \quad (12)$$

Once β'_{new} is determined using (12), the line of flight of the missile during the subsequent time increment δt can be calculated. This is done in effect by computing the equation of the line of flight from the current position of the missile (X_t, Y_t) to the revised aim point. Referring to Figure 6, the general equation for this line (in the horizontal plane) is

$$Y = mX + b \quad (13)$$

The slope m is given by

$$m = \tan (\beta'_{new} + \gamma') \quad (14)$$

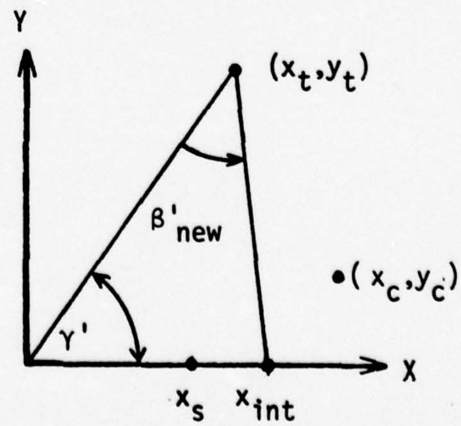

FIGURE 6

while the y - axis intercept point b is determined by solving (13) with $Y = Y_t$, $X = X_t$ and using (14):

$$b = Y_t - m X_t$$

The X - intercept, which is the revised aim point of the missile, is then found by solving (13) for Y=0:

$$X_{int} = X_t - Y_t \cot (\beta'_{new} + \gamma')$$

The missile is then advanced along the new line of flight, and the ship and chaff cloud positions are incremented. If the missile is within one time increment's distance of either the ship or the decoy, it is then determined whether the missile flight path will take it over the ship. Otherwise, the entire process of calculating the error angle α_D and moving the missile, ship and chaff cloud is repeated. In these subsequent interactions of the tracking loop, the second of equations (11) is modified by substituting X_{int} (the current aim point) for Xs.

The implementation of the above algorithm in computer software requires accounting for the algorithm for computing arctangents in FORTRAN IV and also the formulas for determining the rotation angle of the missile frame and the aim point. Consequently, it was necessary to include three tracking loops to deal with first and second, third, and fourth quadrant missile locations.

IV. EVALUATION CRITERIA

Implementation of the models in software permits calculation of the trajectories of the ship, decoy and missile under a wide range of initial conditions. This in turn provides the information required to determine the relative positions of the objects at any time, and thus to evaluate decoy placement effectiveness. This section describes the methodology for that evaluation.

The simulation program evaluates two aspects of decoy placement: whether the decoy is within the field of view of the missile for a sufficient time to affect it (given that the missile is aimed at the ship), and assuming that this is so, whether the decoy will cause a sufficient error in the missile to cause it to miss the ship. It is emphasized from the start that the second of these criteria cannot be as reliable as the first, because it is based on the assumption that track is shifted to the decoy (and not, for example, to the ship-decoy centroid) at some specific time. Nonetheless, the second placement criterion provides an indication of the efficacy of the decoy in causing angle tracking errors.

The first success criterion, called SUCCESS 1 (S_1 for brevity) is based on the purely geometric condition that the decoy is within the missile field of view for a period of time equal to or greater than the decision time of the missile. In the simulation, time is stepped in fixed increments of δt which may or may not be equal to the decision time T_D at the option of the user. To relate δt to T_D, a parameter called ITS is defined as the smallest integer satisfying the following relationship:

$$ITS > \frac{T_D}{\delta t}$$

Then, for example, if $T_D = \delta t$ ITS=2, so that the decoy must be in the field of view at two successive steps in time to count as a success with respect to S_1. The implementation of this condition is shown in Figure 7, which indicates the structure for determining the outcome of a single engagement in the simulation.

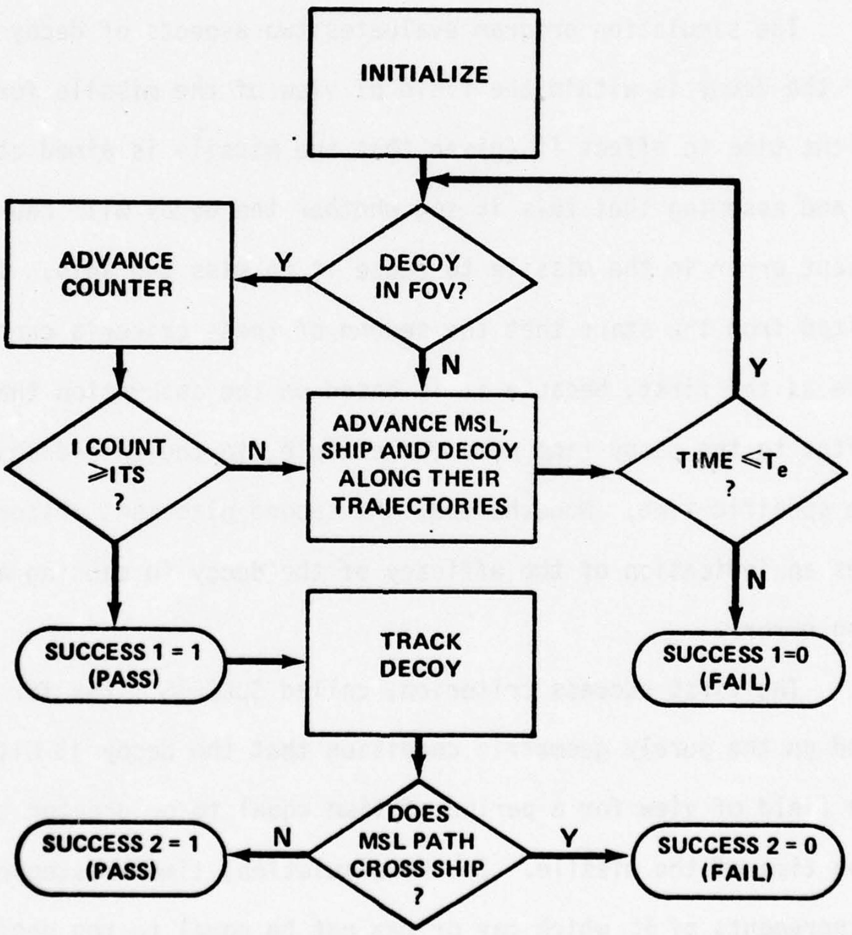

FIGURE 7

Determination of decoy presence within the missile field of view (FOV) is accomplished by evaluating two quantities referenced to the missile: the angle from the ship midpoint to the most distant

cross-range extremity of the decoy, and the extreme distance from the ship midpoint to the decoy boundary along the missile line of flight. If the ship coordinates in the missile frame are (X_S'', Y_S'', Z_S'') while those of the decoy are (X_D'', Y_D'', Z_D''), then the angle CCANG is found be solving the triangle of Figure 8 for α_{DS} using the law of cosines:

$$\text{CCANG} = \cos^{-1}\left(\frac{S^2 + D^2 - A^2}{2SD}\right) \quad (15)$$

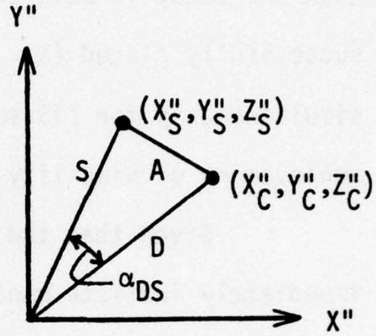

FIGURE 8

where

$$S^2 = X_S''^2 + Y_S''^2 + Z_S''^2$$

$$D^2 = X_C''^2 + Y_C''^2 + Z_C''^2$$

$$A^2 = (X_S'' - X_C'')^2 + (Y_S'' - Y_C'')^2 + (Z_S'' - Z_C'')^2$$

To the angle given by equation (12) is added the angle subtended by half of the decoy as viewed by the missile. *If DCC is the radius of the* chaff cloud, this angle is

$$\text{DANG} = \tan^{-1}(\text{DCC}/D)$$

That is,

$$\text{ANGLE} = \text{CCANG} + \text{DANG}$$

If

$$\text{ANGLE} \leq \text{BWHA} \quad (16)$$

where BWHA is the angle from missile boresight to its 3dB one-way antenna radiation pattern, then the decoy is considered to be within the missile acceptance angle.

The range difference between the ship midpoint and the decoy center as viewed by the missile is

$$\text{PRCC} = |S - D|$$

To this we add DCC, so that the extreme range difference between the midpoint of the ship and the decoy is

$$\text{RANGE} = \text{PRCC} + \text{DCC}$$

Now if,

$$\text{RANGE} \le \text{RGHW} \tag{17}$$

where RGHW is the half-width of the missile range gate, then the decoy is within the missile range gate. If (16) and (17) hold simultaneously, then the decoy is said to be in the field of view (FOV). The decoy is successfully placed (S_1 is true) if equations (16) and (17) hold simultaneously for ITS successive evaluations subsequent to the decoy's achievement of viability (time $t > t_b + t_f$).

Given that the decoy satisfies S_1, the missile is assumed to immediately initiate maneuver toward the decoy. If the flight of the missile toward the decoy, projected in the direction of flight immediately prior to decoy impact or as the missile passes the ship, crosses any part of the ship, it is assumed that the missile will hit the ship. This is evaluated by checking the signs of the Y" coordinates of the ship's bow and stern. If they are both of the same sign, i.e.

$$\begin{gathered} Y''_{SB} > 0 \text{ and } Y''_{SS} > 0 \\ \text{or} \\ Y''_{SB} < 0 \text{ and } Y''_{SS} < 0 \end{gathered} \tag{18}$$

then the second success criterion, SUCCESS 2(or S_2) is satisfied.

In summary: S_2 is true provided that equation (18) holds at the time the missile impacts the decoy or passes the ship.

V. SIMULATION SOFTWARE

The Decoy Placement Simulation RFDPS was written in the version of the FORTRAN IV (G) language utilized by the DEC System 10 computer (DEC, 1972). The program was designed to facilitate its use with time-shared computer facilities, in particular those of TYMSHARE, Inc. Although FORTRAN IV (G) is a standard language and time sharing computing procedures and capabilities are fairly uniform, it is impossible to guarantee that the program described in this section can be utilized on other systems without alteration.

In order to execute RFDPS, it is necessary to compile the program and create two input data files. The methods for accomplishing this using TYMSHARE facilities are described in detail in XEXEC (TYMSHARE, 1974) and EDITOR (TYMSHARE, 1969), so this section will concentrate on the file contents rather than the method for their creation.

The first of the input data files is referred to as the parameter file. It contains the parameters listed in Table 1; the layout is as shown in Table 2. In accordance with good programming style (Kernighan and Plauger, 1974) all parameter names consist entirely of alphabetical characters.

As indicated in Table 1, three seed numbers are required as input parameters; they are used in generating the true wind direction and speed, and the missile initial direction. Three seeds are required because, as shown by MacLaren and Marsaglia (1965) the multiplicative congruential generator cannot produce uniformly distributed triples (See also McQuay (1973)). Since the seeds and the multiplier should be relatively prime with respect to one another, all three seed numbers

NAME	EXPLANATION	RESTRICTIONS
IRPT	NUMBER OF CASES TO BE INCLUDED IN DETAILED REPORT	INTEGER $\leq$ IMMX
IC	VERSION IC = 0 GIVES MONTE CARLO RUN IC $\neq$ 0 GIVES DETERMINISTIC RUN	INTEGER
IMMX	NUMBER OF CASES IN RUN (IC = 0) NUMBER OF THREAT BEARINGS PER VALUE OF VW AND THETA (IC $\neq$ 0)	INTEGER $\leq$ 9999
ISEED (I)	SEEDS FOR RANDOM NUMBER GENERATORS FOR VW, THETA AND GAMMA, RESPECTIVELY	PRIME, OF FORM 8N $\pm$ 3 FOR SOME INTEGER N
VWL	MAXIMUM WIND SPEED, F/S (MONTE CARLO VERSION)	INTEGER $\leq$ 100
DELTAT	TIME INCREMENT BETWEEN SUCCESSIVE EVALUATIONS OF PLACEMENT, SECONDS	REAL > 0
GAMZRO	MINIMUM VALUE OF THREAT DIRECTION, DEG	0 $\leq$ GAMZRO $\leq$ GAMMAX
GAMMAX	MAXIMUM VALUE OF THREAT DIRECTION, DEG	GAMZRO $\leq$ GAMMAX $\leq$ 360°
TF	DECOY TIME OF FLIGHT, S	REAL > 0
TB	DECOY TIME OF SIGNATURE DEVELOPMENT, S	REAL $\geq$ 0
BSHD	HALF LENGTH OF SHIP, F	REAL > 0
RCH	HORIZONTAL RANGE OF DECOY, F	REAL > 0
DCC	RADIUS OF DECOY, F	REAL $\geq$ 0
HZERO	ALTITUDE OF DECOY AT TIME OF BURST, F	REAL $\geq$ 0
VCV	FALL RATE OF DECOY, F/S	REAL
DISTL	LOCATION OF DECOY REFERENCED TO AMIDSHIPS, F	REAL
VSKN	SHIP SPEED, KNOTS	INTEGER $\geq$ 0
IPHI	LAUNCHER TRAIN ANGLE REFERENCED TO SHIP LONGITUDINAL AXIS	0 $\leq$ INTEGER $\leq$ 180°
RZERO	MISSILE RANGE AT TIME OF DECOY LAUNCH, F	REAL > 0
ZTZERO	MISSILE ALTITUDE, F	REAL > 0
VT	MISSILE SPEED, F/S	REAL > 0
YKAPA	MISSILE NAVIGATIONAL CONSTANT, S^{-1}	REAL > 0
BWHA	MISSILE 3 dB HALF BEAMWIDTH	REAL > 0
RGHW	MISSILE RANGE GATE HALF WIDTH, NS	REAL > 0
ITS	NUMBER OF TIMES DECOY MUST BE IN FOV	INTEGER > 0
VWZRO	FOR IC $\neq$ 0, THE MINIMUM WIND SPEED, KN	REAL $\geq$ 0
VWI	FOR IC $\neq$ 0, THE WIND SPEED INCREMENT, KN	POSITIVE REAL SUCH THAT VWM = VWZRO + KVWI FOR SOME INTEGER K
VWM	FOR IC $\neq$ 0, THE MAXIMUM WIND SPEED, KN	REAL $\geq$ VWO
THZRO	FOR IC $\neq$ 0, THE MINIMUM TRUE WIND DIRECTION, DEG	REAL, 0 $\leq$ THZRO $\leq$ 360°
THETI	FOR IC $\neq$ 0, THE TRUE WIND DIRECTION INCREMENT, DEG	POSITIVE REAL SUCH THAT THETM + THZRO + NTHETI FOR SOME INTEGER n
THETM	FOR IC $\neq$ 0, THE MAXIMUM TRUE WIND DIRECTION, DEG	REAL $\geq$ THETO

TABLE 1

should be chosen to be prime. Further, since the period of the generator is maximized (and equal to 2^{b-2} where b is the word length of the computer) for ISEED=8N$\pm$3 where N is an integer (Newman and Odell, 1971), the seed numbers should all be of that form. Such is always possible, and can be accomplished with little effort (Carmichael, 1914). A partial table of good seed numbers, drawn from Abramowitz and Stegun (1973), is included as Appendix B.

The second of the input files contains the true wind speed distribution table, which is discussed in the Appendix. The table DIST contains 100 pairs of four digit integers (the program will accept 150 pairs without modification). The first entry is the true wind in feet per second; the second is a four digit number which is used in conjunction with the random number generator for the true wind speed to obtain a sample population resembling the desired parent population.

The program can be executed using DO loops to generate VW (true wind speed) and THETA (true wind direction) in addition to the Monte Carlo method of generation. This mode of operation can be of great utility when a limited range of wind conditions is of interest, as when evaluating decoy placement against a scenario which is specific in terms of geographic or weather conditions and ship direction of advance.

To run the program using TYMSHARE, the following sequence of operations is required. The sequence below assumes that the data files have been written and the program compiled. Underlined statements are entered by the user; the symbol ¢ indicates a carriage return.

(Log on to system)

- <u>EXECUTE RFDPS ¢</u>

LOADING

EXECUTION

ENTER PARAM FILE NAME: <u>PARAMF¢</u>

ENTER NAME OF DIST TABLE: <u>DIST¢</u>

ENTER SUM RPT FILE NAME: <u>SUMARY¢</u>

ENTER DTL RPT FILE NAME: <u>DETAIL¢</u>

(Program Executes)

TOTAL TRU FOR RUN=XXX

TRU PER CASE=YYY

EXIT

- (Log off from system or repeat sequence for another data set)

In the above example, the file PARAMF contained the parameters listed in Table 2, DIST was the table of true wind speed probabilities; the summary file SUMARY and the detail (case by case) file DETAIL were written. The last two lines gave the CPU use for the run and the CPU use per case.

If desired, the output files can be typed on the remote terminal simply by giving the command TYPE (File Name) ¢. Alternatively, they can be listed (at lower cost) on a remote high speed printer by means of the SPOOL command. Both files are width compatible with TTY size paper and half width terminals.

The remaining pages of this section contain complete listings for RFDPS and DIST. The main program requires approximately 15.3k bytes of core in the compiled version DIST requires an additional 0.9k bytes.

FIRST LINE: FORMAT (7I, 3F) "PROGRAM CONTROL"

IRPT, IMMX, IC, ISEED (1), ISEED (2), ISEED (3), VWL, DELTAT, GAMZRO, GAMMAX

SECOND LINE: FORMAT (8F, 2I) "DECOY SYSTEM/INSTALLATION AND SHIP PARAMETERS"

TF, TB, BSHD, RCH, DCC, HZERO, VCV, DISTL, VSKN, IPHI

THIRD LINE: FORMAT (6F, I) "MISSILE PARAMETERS"

RZERO, ZTZERTO, VT, YKAPA, BWHA, RGHW, ITS

FOURTH LINE: FORMAT (10A4) "FILE NAME"

FILE NAME

FIFTH LINE: FORMAT (6F) "WIND PARAMETERS"

VWZRO, VWI, VWM, THETZRO, THETI, THETM

TABLE 2

RFDPS LISTING

```
      DOUBLE PRECISION NAME,DNAME,SNAME,FNAME
      INTEGER SUCS1,SUCS2,VWL,VSKN
      DIMENSION IVWT(2,150),ISEED(3),IG(24),IGF(2,24),
     -GS(2,24),IRW(24),IRWF(2,24),RWS(2,24),IWV(13),
     -IVWF(2,13),VWS(2,13),IGLO(24),IGHI(24),
     -IVLO(12),IVHI(12),IDLO(24),IDHI(24),IDSC(10)
      XSTRT=TRU(XXX)
C***  READ IN PARAM CARDS
      TYPE 4
    4 FORMAT(' ENTER PARAM FILE NAME: ',$)
      ACCEPT 7,FNAME
    7 FORMAT(A10)
      OPEN(8,FNAME,INPUT,SYMBOLIC,ERR=5)
      GO TO 10
    5 TYPE 6
    6 FORMAT('NO PARAM FILE--PROGRAM ABORTED.')
      STOP
   10 READ (8,11) IRPT,IMMX,IC,ISEED,VWL,DELTAT,GAMZRO,GAMMAX
   11 FORMAT(7I,3F)
      READ(8,12) TF,TB,BSHD,RCH,DCC,HZERO,VCV,DISTL,VSKN,IPHI
   12 FORMAT(8F,2I)
      READ(8,13) RZERO,ZTZERO,VT,YKAPA,BWHA,RGHW,ITS
   13 FORMAT(6F,I)
      READ(8,14) (IDSC(I),I=1,10)
   14 FORMAT(10A4)
      READ(8,15)VWZRO,VWI,VWM,THZRO,THETI,THETM
   15 FORMAT(6F)
      CLOSE(8)
C***  READ IN VW DISTRIBUTION TABLE.
      I=1
      TYPE 18
   18 FORMAT(' ENTER FILE NAME OF DIST TABLE: ',$)
      ACCEPT 7,NAME
      OPEN(9,NAME,INPUT,SYMBOLIC,ERR=21)
      GO TO 19
   21 TYPE 17
   17 FORMAT(' NO VW DISTRIBUTION TABLE - PROG. ABORTED.')
      STOP
   19 READ (9,16,END=20) IVWT(1,I),IVWT(2,I)
   16 FORMAT(2I4)
      I=I+1
      IF(I.GT.150) GO TO 20
      GO TO 19
   20 IVTMX=I-1
      CLOSE(9)
      DO 25 J=1,IVTMX
      IF(VWL.LE.IVWT(1,J)) GO TO 30
   25 CONTINUE
   30 IMAX=IVWT(2,J)
C*** INITILIZE COUNTERS
```

```
      ITHM=(THETM-THZRO)/THETI+1.0
      IVWM=(VWM-VWZRO)/VWI+1.0
C*** OPEN I/O FILES.
      TYPE 37
   37 FORMAT(' ENTER SUM RPT FILE NAME: ',$)
      ACCEPT 7,SNAME
      OPEN(10,SNAME,OUTPUT,SYMBOLIC)
      IF(IRPT.GT.0) TYPE 38
   38 FORMAT(' ENTER DTL RPT FILE NAME: ',$)
      IF(IRPT.GT.0) ACCEPT 7,DNAME
      IF(IRPT.GT.0) OPEN(11,DNAME,OUTPUT,SYMBOLIC)
      CRD=57.29578
      CNMF=6076
      VS=VSKN*CNMF/3600
      PHI=IPHI/CRD
      SPHI=SIN(PHI)
      CPHI=COS(PHI)
      BWHA=BWHA/CRD
      RGHW=RGHW*.984
      YDIS=VT*DELTAT
C*****CLEAR ARRAYS
      DO 40 I=1,12
      IG(I)=0
      IWV(I)=0
   40 IWV(13)=0
      DO 45 I=1,24
      IRW(I)=0
   45 IFAIL1=0
      IFAIL2=0
      IFAIL=0
C*****LOOP THRU IM
   50 DO 2375 IM=1,IMMX
C*********************
C     IF DETERMINISTIC, LOOP THROUGH THETA AND VW
C*********************
      IF(IC.EQ.0) GO TO 60
      DO 2350 ITHET=1,ITHM
      THETA=THZRO+(ITHET-1)*THETI
      THETA=THETA/CRD
      DO 2300 IVW=1,IVWM
      VW=VWZRO+(IVW-1)*VWI
      VW=VW*3600/CNMF
      STHETA=SIN(THETA)
      CTHETA=COS(THETA)
   60 CONTINUE
   76 IF(VWL.GT.0.0) GO TO 80
      GO TO 110
C*****GET RANDOM VW
      IF(IC.GT.0.0) GO TO 120
   80 YFL=RAN(ISEED(1))
```

```
      ITEMP=YFL*IMAX+.5
      IF(ITEMP.LT.1) GO TO 80
   85 DO 90 I=1,IVTMX
      IF(ITEMP.GT.IVWT(2,I)) GO TO 90
      GO TO 95
   90 CONTINUE
   95 IF(IVWT(1,I).GT.VWL) GO TO 80
      VW=IVWT(1,I)
      IVW=IVWT(1,I)
      IF(VW.EQ.0.0) GO TO 110
C*** GET RANDOM THETA.
  100 YFL=RAN(ISEED(2))
      K=YFL*379.0
      IF(K.LT.10.OR.K.GT.369) GO TO 100
      ITHETA=K-10
      THETA=ITHETA/CRD
      STHETA=SIN(THETA)
      CTHETA=COS(THETA)
      GO TO 120
  110 ITHETA=0
      THETA=0.0
      STHETA=0.0
      CTHETA=1.0
      VW=0.0
C*** GET RANDOM GAMMA.
  120 YFL=RAN(ISEED(3))
      K=YFL*(GAMMAX+19.0)
      IF(K.LT.(GAMZRO+10.0).OR.K.GT.(GAMMAX+10.0)) GO TO 120
      IGAMMA=K-10
      GAMMA=IGAMMA/CRD
      SGAMMA=SIN(GAMMA)
      CGAMMA=COS(GAMMA)
C*****CALCULATE TEND
      BETA=ASIN(VS*SGAMMA/VT)
      SBG=SIN(BETA+GAMMA)
      CBG=COS(BETA+GAMMA)
      IF(IGAMMA.NE.0.AND.IGAMMA.NE.180) GO TO 190
      IF(IGAMMA.EQ.180) TEND=RZERO/(VT-VS)
      IF(IGAMMA.EQ.0) TEND=RZERO/(VT+VS)
      GO TO 191
 190  TEND=RZERO*SGAMMA/(VT*SBG)
C*** CALCULATE RWA
  191 IRWA=0.0
      IF(VWL.EQ.0) GO TO 250
  200 TEMP=VW*CTHETA+VS
      IF(ITHETA.EQ.180) GO TO 210
      IF(TEMP.NE.0.0) GO TO 220
      IF(ITHETA.LT.180) GO TO 205
      ZETA=270.0
      GO TO 240
```

```
  205 ZETA=90.0
      GO TO 240
  210 IF(TEMP.LT.0.0) GO TO 215
      ZETA=0.0
      GO TO 240
  215 ZETA=180.0
      GO TO 240
  220 TEMP=(VW*STHETA)/TEMP
      IF(TEMP.LT.0.0) TEMP=-TEMP
      ZETA=ATAN(TEMP)
      ZETA=ZETA*CRD
      TEMP=VW*CTHETA
      IF(TEMP.LT.0.0) TEMP=-TEMP
      IF(ITHETA.GE.0.AND.ITHETA.LE.90) GO TO 240
      IF(ITHETA.LE.180) GO TO 230
      IF(ITHETA.LE.270) GO TO 235
  225 ZETA=360-ZETA
      GO TO 240
  230 IF(TEMP.LE.VS) GO TO 240
      ZETA=180-ZETA
      GO TO 240
  235 IF(TEMP.LE.VS) GO TO 225
      ZETA=ZETA+180
  240 IRWA=ZETA
C*** CALCULATE VRW
  250 VRW=SQRT(VW*VW+VS*VS+2*VW*VS*CTHETA)
C*** INITILIZE TIME.
  300 T=TB+TF
      P1=RCH*CPHI+DISTL+VS*TF
      P2=VW*CTHETA
      P3=RCH*SPHI
      P4=VW*STHETA
      P5=RZERO*SGAMMA
      P6=RZERO*CGAMMA
      ICNTS=0
      SUCS1=0
      SUCS2=0
      TSUCS1=0
C*****CALCULATE COORDS IN EARTH FRAME
  320 XC=P1-T*P2
      YC=P3-T*P4
      ZC=HZERO-VCV*(T-TF)
      IF(ZC.LT.0.0) ZC=0.0
      XS=VS*T
      XT=P6+VS*T-T*P6/TEND
  326 YT=P5*(1-T/TEND)
C*****CALCULATE COORDS IN TRACKER FRAME
      XDS=(XT-XS)*CBG+YT*SBG
      YDS=(XS-XT)*SBG+YT*CBG
      XDSB=(XT-XS-BSHD)*CBG+YT*SBG
```

```
      XDSS=(XT-XS+BSHD)*CBG+YT*SBG
      YDSB=(XS+BSHD-XT)*SBG+YT*CBG
      YDSS=(XS-BSHD-XT)*SBG+YT*CBG
      RHO=SQRT(XDS**2+YDS**2)
      ZT=ZTZERO
 328  ZDS=-ZT
      XDC=(XT-XC)*CBG+(YT-YC)*SBG
      YDC=(XC-XT)*SBG+(YT-YC)*CBG
      ZDC=ZC-ZT
C***  CALCULATE CCANG.
  600 AH=XDS**2+YDS**2+ZDS**2
      BH=XDC*XDC+YDC*YDC+ZDC*ZDC
      CH=(XDS-XDC)*(XDS-XDC)+(YDS-YDC)*(YDS-YDC)+(ZDS-ZDC)**2
      Z3=(AH+BH-CH)/(2*SQRT(AH*BH))
      IF(Z3.GE.1.0) Z3=1.0
      CCANG=ACOS(Z3)
C*****CALCULATE RANGE DIFFERENCE
  800 PRCC=ABS(SQRT(AH)-SQRT(BH))
C*****EVALUATE SUCCESS ONE
  850 DANG=ATAN(DCC/SQRT(BH))
      ANGLE=CCANG+DANG
      RANGE=PRCC+DCC
      IF(ANGLE.LE.BWHA.AND.RANGE.LE.RGHW) ICNTS=ICNTS+1
      IF(ICNTS.GE.ITS) SUCS1=1
      IF(SUCS1.EQ.1) GO TO 890
 810  T=T+DELTAT
      IF(T.LE.TEND) GO TO 320
      GO TO 1000
C***  TRACK DECOY AND EVALUATE SUCS2
 890  TSUCS1=T
      IF(XT.LT.0.0.AND.YT.LT.0.0) GO TO 930
      IF(XT.GT.0.0.AND.YT.LT.0.0) GO TO 960
  895 GAMPR=ATAN(YT/XT)
      IF(GAMPR.LT.0.0) GAMPR=GAMPR+180/CRD
      BETPR=ASIN(XS*SIN(GAMPR)/SQRT((XT-XS)**2+YT**2))
  900 SGBP=SIN(GAMPR+BETPR)
      CGBP=COS(GAMPR+BETPR)
      XDS=(XT-XS)*CGBP+YT*SGBP
      XDC=(XT-XC)*CGBP+(YT-YC)*SGBP
      YDC=(XC-XT)*SGBP+(YT-YC)*CGBP
      IF(XDC.LT.YDIS.OR.XDS.LT.YDIS) GO TO 920
      ALPHD=ATAN(YDC/XDC)
      BETPR=BETPR+YKAPA*ALPHD*DELTAT
      CGBP=COS(GAMPR+BETPR)
      SGBP=SIN(GAMPR+BETPR)
      XINT=XT-YT*CGBP/SGBP
      XT=XT-YDIS*CGBP
      YT=YT-YDIS*SGBP
      XC=XC-P2*DELTAT
      YC=YC-P4*DELTAT
```

```
  910 XS=XS+VS*DELTAT
      GAMPR=ATAN(YT/XT)
      IF(GAMPR.LT.0.0) GAMPR=GAMPR+180/CRD
      BETPR=ASIN(XINT*SIN(GAMPR)/SQRT((XT-XINT)**2+YT**2))
      GO TO 900
  920 YDSB=(XS+BSHD-XT)*SGBP+YT*CGBP
      YDSS=YDSB-2*BSHD*SGBP
      IF(YDSB.GT.0.0.AND.YDSS.GT.0.0.OR.YDSB.LT.0.0.AND.
     -YDSS.LT.0.0) GO TO 990
      GO TO 980
C*****TRACKING LOOP: 3RD QUADRANT
  930 GAMPR=ATAN(YT/XT)
      BETPR=ASIN(XS*SIN(GAMPR)/SQRT((XT-XS)**2+YT**2))
  935 SGBP=SIN(GAMPR-BETPR)
      CGBP=COS(GAMPR-BETPR)
      XDC=(XC-XT)*CGBP+(YC-YT)*SGBP
      YDC=(XT-XC)*SGBP+(YC-YT)*CGBP
      XDS=(XS-XT)*CGBP-YT*SGBP
      IF(XDC.LT.YDIS.OR.XDS.LT.YDIS.OR.YT.GT.-YDIS) GO TO 940
      ALPHD=ATAN(YDC/XDC)
      BETPR=BETPR-YKAPA*ALPHD*DELTAT
      CGBP=COS(GAMPR-BETPR)
      SGBP=SIN(GAMPR-BETPR)
      XINT=XT-YT*CGBP/SGBP
      XT=XT+YDIS*CGBP
      YT=YT+YDIS*SGBP
      XC=XC-P2*DELTAT
      YC=YC-P4*DELTAT
      XS=XS+VS*DELTAT
      GAMPR=ATAN(YT/XT)
      BETPR=ASIN(XINT*SIN(GAMPR)/SQRT((XT-XINT)**2+YT**2))
      IF(XT.LT.0.0.AND.YT.LT.0.0) GO TO 935
      IF(XT.GT.0.0.AND.YT.LT.0.0) GO TO 960
      GO TO 895
  940 YDSB=(XT-XS-BSHD)*SGBP+YT*CGBP
      YDSS=YDSB+2*BSHD*SGBP
      IF(YDSB.GT.0.0.AND.YDSS.GT.0.0.OR.YDSB.LT.0.0.AND.
     -YDSS.LT.0.0) GO TO 990
      GO TO 980
C*****TRACKING LOOP: 4TH QUADRANT
  960 GAMPR=-ATAN(YT/XT)
      BETPR=ASIN(XS*SIN(GAMPR)/SQRT((XT-XS)**2+YT**2))
  965 SGBP=SIN(GAMPR+BETPR)
      CGBP=COS(GAMPR+BETPR)
      XDC=(XT-XC)*CGBP+(YC-YT)*SGBP
      YDC=(XT-XC)*SGBP+(YT-YC)*CGBP
      XDS=(XT-XS)*CGBP-YT*SGBP
      IF(XDC.LT.YDIS.OR.XDS.LT.YDIS.OR.YT.GT.-YDIS) GO TO 970
      ALPHD=ATAN(YDC/XDC)
      BETPR=BETPR-YKAPA*ALPHD*DELTAT
```

```
      CGBP=COS(GAMPR+BETPR)
      SGBP=SIN(GAMPR+BETPR)
      XINT=XT+YT*CGBP/SGBP
      XT=XT-YDIS*CGBP
      YT=YT+YDIS*SGBP
      XC=XC-P2*DELTAT
      YC=YC-P4*DELTAT
      XS=XS+VS*DELTAT
      GAMPR=-ATAN(YT/XT)
      BETPR=ASIN(XINT*SIN(GAMPR)/SQRT((XT-XINT)**2+YT**2))
      IF(XT.GT.0.0.AND.YT.LT.0.0) GO TO 965
      IF(XT.LT.0.0.AND.YT.LT.0.0) GO TO 930
      GO TO 895
  970 YDSB=(XT-XS-BSHD)*SGBP+YT*CGBP
      YDSS=YDSB+2*BSHD*SGBP
      IF(YDSB.GT.0.0.AND.YDSS.GT.0.0.OR.YDSB.LT.0.0.AND.
     -YDSS.LT.0.0) GO TO 990
  980 SUCS2=0
      GO TO 1000
  990 SUCS2=1
C*** WRITE DETAIL REPORT
 1000 CONTINUE
      IGAMMA=GAMMA*CRD
      IVRW=VRW*3600/CNMF+.5
      IF(IM.GT.IRPT) GO TO 2010
      IF(IM.GE.2) GO TO 1100
      WRITE(11,1050) FNAME,VSKN,IPHI
 1050 FORMAT(1X,A10,2X,'DETAILED REPORT FOR VS = ',I2,
     -'KNOTS AND PHI = ',I3,'DEGREES'/
     -4X,'GAMMA',5X,'RWA',5X,'VRW',2X,'SUCS1',2X,'TSUCS1',2X,'SUCS2'/
     -3X,'DEG.REL.  DEG.REL.  KN',10X,'SEC.'//)
 1100 WRITE(11,1150) IGAMMA,IRWA,IVRW,SUCS1,TSUCS1,SUCS2
 1150 FORMAT(5X,I3,6X,I3,5X,I3,3X,I1,5X,F4.1,5X,I1)
C***  ACCUMULATE RUN STATISTICS
 2010 IF(SUCS1.EQ.0) IFAIL1=IFAIL1+1
      IF(SUCS1.EQ.1.AND.SUCS2.EQ.0) IFAIL2=IFAIL2+1
      IF(SUCS1.EQ.0.OR.SUCS1.EQ.1.AND.SUCS2.EQ.0) IFAIL=IFAIL+1
C*** ACCUMULATE THREAT GEOMETRY AND REL. WIND ANGLE STATISTICS
      DO 2050 I=1,24
      M=15*(I-1)
      N=15*I
      IF(M.LE.IGAMMA.AND.N.GT.IGAMMA) IG(I)=IG(I)+1
      IF(M.LE.IGAMMA.AND.N.GT.IGAMMA.AND.SUCS1.EQ.0)
     -IGF(1,I)=IGF(1,I)+1
      IF(M.LE.IGAMMA.AND.N.GT.IGAMMA.AND.SUCS2.EQ.0.
     -AND.SUCS1.EQ.1) IGF(2,I)=IGF(2,I)+1
 2050 CONTINUE
      DO 2100 I=1,24
      M=15*(I-1)
      N=15*I
```

```
      IF(M.LE.IRWA.AND.N.GT.IRWA) IRW(I)=IRW(I)+1
      IF(M.LE.IRWA.AND.N.GT.IRWA.AND.SUCS1.EQ.0)
     -IRWF(1,I)=IRWF(1,I)+1
      IF(M.LE.IRWA.AND.N.GT.IRWA.AND.SUCS2.EQ.0.
     -AND.SUCS1.EQ.1) IRWF(2,I)=IRWF(2,I)+1
 2100 CONTINUE
      DO 2200 I=1,12
      M=5*(I-1)
      N=5*I
      IF(M.LE.IVRW.AND.N.GT.IVRW) IWV(I)=IWV(I)+1
      IF(M.LE.IVRW.AND.N.GT.IVRW.AND.SUCS1.EQ.0)
     -IVWF(1,I)=IVWF(1,I)+1
      IF(M.LE.IVRW.AND.N.GT.IVRW.AND.SUCS2.EQ.0.
     -AND.SUCS1.EQ.1) IVWF(2,I)=IVWF(2,I)+1
 2200 CONTINUE
      IF(IVRW.LT.60) GO TO 2250
      IWV(13)=IWV(13)+1
      IF(SUCS1.EQ.0) IVWF(1,13)=IVWF(1,13)+1
      IF(SUCS2.EQ.0) IVWF(2,13)=IVWF(2,13)+1
 2250 IF(IC.EQ.0.0) GO TO 2375
C***   END VW LOOP
 2300 CONTINUE
C***   END THETA LOOP
 2350 CONTINUE
C*** END IM LOOP
 2375 CONTINUE
C*** WRITE SUMMARY REPORT
 2400 DO 2450 I=1,24
      IF(IG(I).NE.0) GO TO 2425
      GS(1,I)=0
      GS(2,I)=0
      GO TO 2450
 2425 GS(1,I)=FLOAT(IG(I)-IGF(1,I))/IG(I)
      IF(IG(I).EQ.IGF(1,I)) GO TO 2450
      GS(2,I)=FLOAT(IG(I)-IGF(2,I)-IGF(1,I))/(IG(I)-IGF(1,I))
 2450 CONTINUE
      DO 2500 I=1,24
      IF(IRW(I).NE.0) GO TO 2475
      RWS(1,I)=0
      RWS(2,I)=0
      GO TO 2500
 2475 RWS(1,I)=FLOAT(IRW(I)-IRWF(1,I))/IRW(I)
      IF(IRW(I).EQ.IRWF(1,I)) GO TO 2500
      RWS(2,I)=FLOAT(IRW(I)-IRWF(2,I)-IRWF(1,I))/(IRW(I)-IRWF(1,I))
 2500 CONTINUE
      DO 2600 I=1,13
      IF(IWV(I).NE.0) GO TO 2550
      VWS(1,I)=0
      VWS(2,I)=0
      GO TO 2600
```

```
2550 VWS(1,I)=FLOAT(IWV(I)-IVWF(1,I))/IWV(I)
     IF(IWV(I).EQ.IVWF(1,I)) GO TO 2600
     VWS(2,I)=FLOAT(IWV(I)-IVWF(2,I)-IVWF(1,I))/(IWV(I)-IVWF(1,I))
2600 CONTINUE
     SR1=FLOAT(IMMX-IFAIL1)/IMMX
     IF(IFAIL1.EQ.IMMX) GO TO 2650
     SR2=FLOAT(IMMX-IFAIL2-IFAIL1)/(IMMX-IFAIL1)
2650 SR3=FLOAT(IMMX-IFAIL1-IFAIL2)/IMMX
     BSHD2=2*BSHD
     WRITE(10,2700) FNAME
2700 FORMAT(15X,'INPUT FILE:',2X,A10)
     WRITE(10,2710) VSKN
2710 FORMAT(15X,'SHIP SPEED:',4X,I2,1X,'KNOTS')
     WRITE(10,2720) IPHI
2720 FORMAT(2X,'LAUNCHER POINTING ANGLE:',3X,I3,1X,'DEG REL')
     WRITE(10,2730) BSHD2
2730 FORMAT(14X,'SHIP LENGTH:',3X,F4.0,1X,'FEET')
     WRITE(10,2740) DISTL
2740 FORMAT(8X,'LAUNCHER LOCATION:',2X,F5.1,1X,'FT FM CTR'/)
     WRITE(10,2750) IMMX
2750 FORMAT(2X,'RUN STATISTICS:  TOTAL NUMBER OF RUNS  :  ',I4)
     WRITE(10,2760) IFAIL1
2760 FORMAT(19X,'TOTAL NUMBER OF FAIL1 :  ',I4)
     WRITE(10,2770) SR1
2770 FORMAT(19X,'TYPE 1 SUCCESS RATIO  :  ',F4.2)
     WRITE(10,2780) IFAIL2
2780 FORMAT(19X,'TOTAL NUMBER OF FAIL2 :  ',I4)
     WRITE(10,2790) SR2
2790 FORMAT(19X,'TYPE 2 SUCCESS RATIO  :  ',F4.2)
     WRITE(10,2795) SR3
2795 FORMAT(19X,'CENTROID SUCCESS RATIO:  ',F4.2/)
     WRITE(10,2800)
2800 FORMAT(6X,'FAILURES BY TRACKER DIRECTION'/
    -2X,'DIRECTION',2X,'NO.OF',4X,'FAIL1',7X,'FAIL2'/
    -2X,'(DEG REL)',2X,'CASES',2X,'NO.',2X,'FRAC.',2X,'NO.',
    -2X,'FRAC.'//)
     DO 2820 I=1,24
     IGLO(I)=15*(I-1)
     IGHI(I)=15*I
     WRITE(10,2810) IGLO(I),IGHI(I),IG(I),IGF(1,I),GS(1,I),
    -IGF(2,I),GS(2,I)
2810 FORMAT(3X,I3,'-',I3,3X,I4,2X,I4,2X,F4.2,2X,I4,2X,F4.2)
2820 CONTINUE
     WRITE(10,2840)
2840 FORMAT(//3X,'FAILURES BY RELATIVE WIND DIRECTION'/
    -2X,'DIRECTION  NO.OF',4X,'FAIL1',7X,'FAIL2'/
    -2X,'(DEG REL)  CASES  NO.  FRAC.  NO.  FRAC.'//)
     DO 2860 I=1,24
     IDLO(I)=15*(I-1)
     IDHI(I)=15*I
```

```
      WRITE(10,2850) IDLO(I),IDHI(I),IRW(I),IRWF(1,I),RWS(1,I),
     -IRWF(2,I),RWS(2,I)
 2850 FORMAT(3X,I3,'-',I3,3X,I4,2X,I4,2X,F4.2,2X,I4,2X,F4.2)
 2860 CONTINUE
      WRITE(10,2870)
 2870 FORMAT(//4X,'FAILURES BY RELATIVE WIND SPEED'/
     -3X,'SPEED  NO.OF',4X,'FAIL1',7X,'FAIL2'/
     -3X,'(KN)',3X,'CASES   NO.   FRAC.   NO.   FRAC.'//)
      DO 2890 I=1,12
      IVLO(I)=5*(I-1)
      IVHI(I)=5*I
      WRITE(10,2880) IVLO(I),IVHI(I),IWV(I),IVWF(1,I),VWS(1,I),
     -IVWF(2,I),VWS(2,I)
 2880 FORMAT(3X,I2,'-',I2,2X,I4,2X,I4,2X,F4.2,2X,I4,2X,F4.2)
 2890 CONTINUE
      WRITE(10,2900) IWV(13),IVWF(1,13),VWS(1,13),IVWF(2,13),VWS(2,13)
 2900 FORMAT(3X,'GE 60',2X,I4,2X,I4,2X,F4.2,2X,I4,2X,F4.2/)
C######
      XNOW=TRU(XXX)-XSTRT
      TYPE 4006,XNOW
 4006 FORMAT(' TOTAL TRU FOR RUN = ',F8.4)
      XNO=XNOW/IMMX
      TYPE 4007,XNO
 4007 FORMAT(' TRU PER CASE = ',F10.7)
C#########
      STOP
      END
```

```
DIST          PAGE ONE

 0    1
 1    4
 2   11
 3   27
 4   54
 5   97
 6  159
 7  243
 8  349
 9  479
10  632
11  808
121006
131224
141460
151712
161978
172255
182541
192834
203132
213432
223733
234033
244330
254623
264910
275191
285465
295729
305985
316231
326468
336695
346911
357117
367313
377498
387674
397840
407996
418143
428281
438411
448532
458645
468751
478850
488942
499027
```

DIST PAGE TWO

509106
519179
529247
539310
549368
559422
569472
579518
589560
599599
609635
619668
629698
639725
649750
659773
669794
679813
689831
699847
709862
719898
729887
739898
749908
759917
769925
779933
789940
799946
809952
819957
829962
839966
849970
859973
869976
879979
889982
899984
909986
919988
929990
939992
949993
959994
969995
979996
989997
999998

VI. REFERENCES

Abramowitz, M. and I.A. Stegun, eds. 1972 Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. IBSN 486-61272-4. New York.

Carmichael, R.D. 1914. The Theory of Numbers and Diophantine Analysis. New York.

Digital Equipment Corporation. May 1972. DEC System 10 FORTRAN IV Programmer's Reference Manual. DEC-10-AFDO-D Maynard, M.A.

Kernighan, B.W. and P.J. Plauger. 1974. The Elements of Programming Style. IBSN 0.07-034199-0. New York

Long, M.W., R.D. Wetherington, J.L. Edwards and A.B. Abeling 1965. Wavelength Dependence of Sea Echo. Astia Document 477 905. Georgia Institute of Technology

MacLaren, M.D. and G. Marsaglia 1965. Uniform Random Number Generators. J.ACM 12.

McQuay, W.K. 1973. Computer Simulation Methods for Military Operations Research. Astia Document 771 614. Air Force Avionics Laboratory

Newman, T.G. and P.L. Odell 1971. The Generation of Random Variates. London

Schiff, S.C., R.F. Owens and J.C. Potts. 1975. Chaff Reference Manual. (Unpublished) SCI Report 1156-040. Systems Consultants, Inc.

Schiff, S.C. 1977. Analysis of Surface IR Decoy Placement for Antiship Missile Defense; Volume I: Study Methodology Astia Document A039 308. Systems Consultants, Inc.

TYMESHARE, Inc. 1969 EDITOR TYMSHARE Reference Manual. Cupertino, CA.

TYMSHARE, Inc. 1974 XEXEC TYMSHARE Manual TYMCOM-X. Cupertino, CA.

APPENDIX A

THE DISTRIBUTION OF TRUE WIND SPEEDS AT SEA AND AN ALGORITHM FOR CALCULATING RELATIVE WIND

A.1 DISTRIBUTION OF TRUE WIND SPEEDS

The decoy placement simulations described in this volume require input data concerning true wind conditions since these, along with the ship's speed, determine the relative motion of a free falling airborne decoy with respect to the ship, and hence also with respect to the missile. Whereas the direction of the true wind can be assumed to be uniformly distributed to good approximation in modeling ensembles of encounters, the true wind speed cannot. Hence, it was desirable to obtain empirical data on true wind speed frequencies of occurence in a maritime environment.

Fortunately, such data were reported by Long and his co-workers (Long, et al, 1965). Their report contained a curve giving true wind speeds and their frequencies of occurrence; the curve resulted from the aggregation of data for a large number of locations in several oceans over the period of many years.

Long's curve was analyzed to determine if it was reasonably well approximated by any known probability density function. The results of the analysis indicated an excellent fit for a slightly modified χ^2 distribution having eight degrees of freedom. The words "slightly modified" are significant here, even though the modification was extremely slight*. It must be emphasized that no claim is being made for a χ^2 distribution with 8 degrees of freedom being the result of some underlying property of nature. We could have done just as well with a polynomial curve fit.

* It consisted of asigning a non-zero probability ($P=10^{-4}$) to VW=0

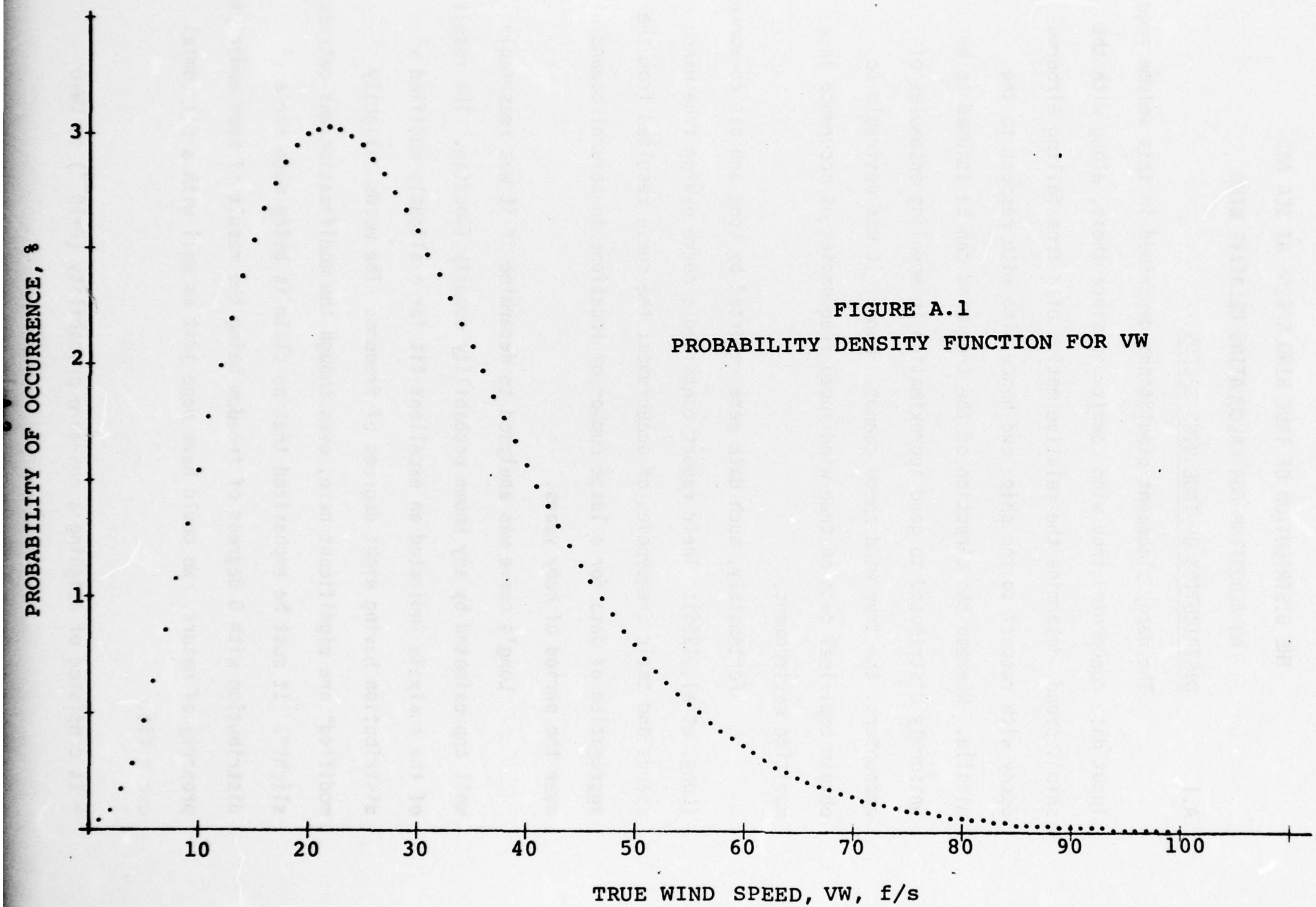

FIGURE A.1

PROBABILITY DENSITY FUNCTION FOR VW

Once a satisfactory fit to Long's curve was obtained, it was quantized. The result, plotted again in the form of a curve, is shown in Figure A1. Arranged in tabular form, the curve yields the table DIST.

A2. CALCULATION OF RELATIVE WIND SPEED AND ANGLE

Given the direction of the wind θ as measured in the earth frame, the wind speed V_W and the ship speed V_S, the relative wind speed V_{RW} and the relative wind direction β can be determined by solving the vector diagram shown in Figure A.2. Using the law of cosines,

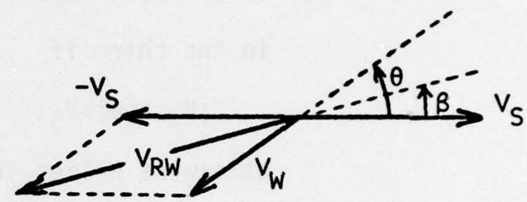

FIGURE A.2

$$V_{RW}=(V^2{}_S+V^2{}_W+2V_WV_S\cos\theta)^{\frac{1}{2}} \qquad \text{A.1}$$

Using the law of sines,

$$\beta=\sin^{-1}(V_W\sin\theta/V_{RW}) \qquad \text{A.2}$$

Now equation (A.1) uniquely specifies V_{RW} for any triple (V_W, V_S, θ); however (A.2) is multivalued: for given V_W, V_{RW} and $\theta \neq N+1)\pi/2$, it admits roots β_1 and β_2 such that $\beta_2=\pi-\beta_1$. Clearly, only one of the roots can be physically meaningful, but ostensibly it could be either one. For example, if θ=0, β will be equal to 0^o if $V_S>V_W$, and equal to 180^o if $V_W>V_S$. The generalization of this observation, along with recognition of the fact that β will always be in the hemicircle containing θ, V_S, and $-V_S$ allows design of the algorithm for calculating the relative wind angle. The rules for calculation of β are as follows:

1) If θ lies in the first quadrant, so does β

2) If θ lies in the second quadrant, β lies in the second quadrant if

$$|V_W \cos\theta| > V_S;$$

otherwise it lies in the first quadrant

3) If θ lies in the third quadrant, β is also in the third if

$$|V_W \cos\theta| > V_S;$$

otherwise β lies in the fourth quadrant

4) If θ lies in the fourth quadrant, β is also in the fourth quadrant.

APPENDIX B

FIVE DIGIT PRIMES OF FORM PRIME = 8N±3

10067	30011	50021	70003	90011
10091	30059	50051	70019	90019
10099	30091	50053	70051	90053
10163	30139	50069	70061	90059
10211	30187	50077	70067	90067
10243	30203	50093	70099	90107
10267	30211	50101	70117	90149
10331	30259	50123	70123	90163
10427	30307	50131	70139	90173
10459	30323	50147	70141	90187
10499	30347	50221	70157	90197
10531	30403	50227	70163	90203
10627	30427	50261	70181	90227
10651	30467	50291	70229	90371
10667	30491	50333	70237	90373
10691	30643	50341	70309	90379
10723	30661	50363	70373	90397
10739	30677	50387	70379	90403
10771	30707	50411	70381	90437
10859	30757	50459	70429	90469
20011	40013	60013	80021	
20051	40037	60029	80051	
20107	40093	60037	80077	
20123	40099	60077	80107	
20147	40123	60083	80141	
20219	40163	60091	80147	
20323	40189	60101	80149	
20347	40213	60107	80173	
20411	40237	60133	80221	
20443	40253	60139	80251	
20483	40277	60149	80309	
20507	40283	60251	80317	
20563	40357	60259	80341	
20611	40387	60293	80347	
20627	40427	60317	80363	
20707	40429	60331	80387	
20731	40459	60373	80429	
20747	40483	60397	80491	
20771	40493	60413	80557	
20899	40499	60427	80603	